

REMARKS/ARGUMENTS

Claims 1-3 and 5-12 are pending in the present application. In the Office Action mailed June 21, 2004, the Examiner rejected claims 1-3 and 5-12 under 35 U.S.C. § 102(b).

Reconsideration is respectfully requested in view of the above amendments to the claims and the following remarks.

A. Claims 1-3 and 5-12 Rejected Under 35 U.S.C. § 102(b)

The Examiner rejected claims 1-3 and 5-12 under 35 U.S.C. § 102(b) as being anticipated by an article entitled "Experience With Secure Multi-Processing In Java" by Dirk Balfanz and Li Gong (hereinafter, "Balfanz"). This rejection is respectfully traversed.

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." MPEP § 2131 (citing Verdegaal Bros. v. Union Oil Co. of California, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987)). "The identical invention must be shown in as complete detail as is contained in the ... claim." Id. (citing Richardson v. Suzuki Motor Co., 9 USPQ2d 1913, 1920 (Fed. Cir. 1989)). In addition, "the reference must be enabling and describe the applicant's claimed invention sufficiently to have placed it in possession of a person of ordinary skill in the field of the invention." In re Paulsen, 31 USPQ2d 1671, 1673 (Fed. Cir. 1994).

Applicants respectfully submit that claims 1-3 and 5-12 are patentably distinct from Balfanz. Balfanz does not disclose all of the limitations in these claims.

Claims 1, 9, and 12 have been amended to recite that several steps that are performed "in response to an object of a first application requesting a resource of a shared base class." Specifically, "a first class loader that loaded the object" is identified. "[I]f the object was loaded by a first class loader that was created for the application," then "information in the first class loader and its associated namespace" is used "to identify application properties." However, "if the object was loaded by the primordial class loader," then "a first thread group created for the first application" is identified and "information associated with the first thread group" is used "to identify the properties

of the first application.” Support for these amendments may be found in page 14, line 7 through page 15, line 7 of Applicants’ specification. Applicants respectfully submit that these amendments further distinguish claims 1, 9, and 12 from Balfanz.

Balfanz does not disclose “identify[ing] application properties” in response to “an object of a[n] application requesting a resource of a shared base class.” Balfanz states the following about applications:

[A]n application has the following properties:

- It is associated with a user that is running the application.
- It holds application-wide state that is shared among all the threads that comprise the application. This state is likely different from that of another application. The state includes:
 - The aforementioned user identification.
 - Distinct standard input, standard output, and error streams.
 - A current working directory.
 - A set of properties.

To implement this concept of an application in Java, we created a class Application based on these criteria and implemented methods to start and stop applications, to query the running user and current directory, query and set the standard streams, and so on.

Balfanz, section 5.1. Thus, Balfanz discloses “a class Application” based on certain criteria, including an “application-wide state that is shared among all the threads that comprise the application.” Id. However, Balfanz does not disclose “identify[ing] application properties” in response to “an object of a[n] application requesting a resource of a shared base class,” as recited in claims 1, 9, and 12.

Balfanz also does not disclose that some objects of an application may be “loaded by a ... class loader that was created for the application” while other objects may be “loaded by the primordial class loader,” as recited in claims 1, 9, and 12. Balfanz states the following about class loaders:

Java uses software-based protection that relies on the type system to provide

basic memory protection and create (by means of class loaders) different name spaces for each application.

In our implementation, every application gets its own copy of the System class. We use a special class loader to re-load and re-define the System class, albeit from the same class material. Since we use a new class loader for every application, to the JVM, the different incarnations of the System class are just different classes that happen to have the same name. This is similar to the case where two applets in a browser may happen to use different classes with the same name. Because the same class are loaded by different class loaders, the underlying JVM treats them as different classes. Now we have a new copy of the System class for every application, we can set the respective System.in, System.out, and System.err streams to point to different things for different applications.

Balfanz, sections 5.1 and 5.5 (emphasis added). Thus, Balfanz discloses “us[ing] a new class loader for every application.” Id. However, Balfanz does not disclose that some objects of an application may be “loaded by a ... class loader that was created for the application” while other objects may be “loaded by the primordial class loader,” as recited in claims 1, 9, and 12.

In view of the foregoing, Applicants respectfully submit that claims 1, 9, and 12 are patentably distinct from Balfanz. Accordingly, Applicants respectfully request that the rejection of claims 1, 9, and 12 be withdrawn.

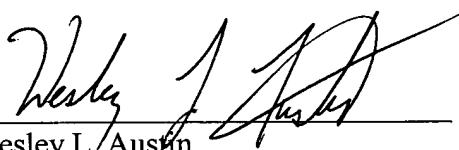
Claims 2-3 and 5-8 depend either directly or indirectly from claim 1. Claims 10-11 depend from claim 9. Accordingly, Applicants respectfully request that the rejection of claims 2-3, 5-8, and 10-11 be withdrawn for at least the same reasons as those presented above in connection with claims 1 and 9.

Appl. No. 09/464,352
Amdt. dated December 20, 2004
Reply to Office Action of June 21, 2004

B. Conclusion

Applicants respectfully assert that all pending claims are patentably distinct from the cited references, and request that a timely Notice of Allowance be issued in this case. If there are any remaining issues preventing allowance of the pending claims that may be clarified by telephone, the Examiner is requested to call the undersigned.

Respectfully submitted,



Wesley L. Austin
Reg. No. 42,273
Attorney for Applicant

Date: December 20, 2004

MADSON & METCALF
Gateway Tower West
15 West South Temple, Suite 900
Salt Lake City, Utah 84101
Telephone: 801/537-1700